Belief Propagation and Locally Bayesian Learning

Adam N. Sanborn (asanborn@gatsby.ucl.ac.uk)
Gatsby Computational Neuroscience Unit, University College London
London, UK

Ricardo Silva (ricardo@stats.ucl.ac.uk)
Department of Statistical Science, University College London
London, UK

Abstract

Highlighting, a conditioning effect, consists of both primacy-like and recency-like effects in human subjects. This combination of effects are notoriously difficult for Bayesian models to produce. An approximation to probabilistic inference, Locally Bayesian learning (LBL), can predict highlighting by partitioning the model into regions during learning and passing messages between these regions. While the approximation matches behavior in this task, it is unclear how LBL compares to other approximations used in Bayesian models, and what behaviors this approximations will predict in other paradigms. Our contribution is to show LBL is closely related to the statistical algorithms of Assumed Density Filtering (ADF), which simplifies calculations by assuming independence, and Belief Propagation, which identifies how to make these calculations through message passing. We propose that people use ADF to learn and show how this model can produce highlighting behavior. In addition, we demonstrate how the degrees of approximation used in LBL and ADF cause the models to make very different predictions in a proposed experimental design.

Keywords: machine learning; conditioning; Bayesian models; belief propagation; assumed density filtering

Bayesian approaches have often been successful in predicting human data as the result of optimal behavior (e.g., Körding & Wolpert, 2004), but people can also produce behavior that is very difficult to explain with Bayesian approaches (Daw, Courville, & Dayan, 2008; Kruschke, 2006a, 2006b). Additionally, Bayesian approaches can be hampered by computational complexity in practical applications (Anderson, 1991).

Modeling human cognition as an approximation to optimal behavior is an approach that has been used to both explain deviations from optimality as well as managing the computational complexity of the solution (Gigerenzer & Todd, 1999; Kahneman, Slovic, & Tversky, 1982; Kruschke, 2006b). Many of these algorithms were invented to match human behavior, but recent work has proposed that candidate algorithms for human cognition could be drawn from work in computer science and statistics (Sanborn, Griffiths, & Navarro, 2006). These algorithms have been developed to efficiently produce results faithful to the full model, and can come with guarantees on the quality of the approximation.

Conditioning has been a testbed for approximations to Bayesian models. Conditioning effects such as blocking and highlighting depend on the order of the stimuli. An approximation that has successfully fit these types of effects is Locally Bayesian Learning (LBL; Kruschke, 2006b). This approximation embodies the compelling idea that local calculations are optimal, but the global prediction is only approximately optimal because information is lost in the communication between regions.

The approximation introduced in LBL was successful in fitting conditioning data, but the approximation itself has not been thoroughly studied. The goal of this paper is to connect the approximation used in LBL with algorithms in computer science and statistics. First we introduce LBL and compare it to the probabilistically correct updating algorithm, Globally Bayesian Learning. Next we introduce Belief Propagation and Assumed Density Filtering (ADF) and show how LBL relates to both these algorithms. Next we describe the effect of highlighting and show that ADF can produce a highlighting effect. Finally we show that LBL predicts a highlighting effect for alternating trials, while ADF does not.

Locally Bayesian Learning

The probabilistic model underlying LBL was a generalization of a feedforward neural network with one hidden layer. This structure and the variable names used are shown in Figure 1a. The neural network was generalized from having single estimates of hidden weights, hidden nodes, and output weights by putting a distribution over the values of these hidden variables. The generalization required different calculations than a neural network to update the weights, and the correct probabilistic update was termed Globally Bayesian Learning (GBL),

$$p(W_{hid}, W_{out}, y \mid x, t) \propto p(t \mid W_{out}, y)\, p(y \mid W_{hid}, x)\, p(W_{out}, W_{hid}) \quad (1)$$

where the variables in the model are described in Figure 1 and $p(t|W_{out}, y)$ and $p(y|W_{hid}, x)$ were the standard linear combination of weights plus a sigmoid nonlinearity.

The network graph was then split into regions to implement LBL, as in Figure 1b. Psychologically, LBL was meant to represent two stages: an early attentional phase that took stimulus cues from the input nodes and converted it into attended cues on the hidden nodes, and weights from the attended cues to the output. Correct probabilistic calculations were used within regions, but LBL uses messages between regions that result in an approximation to GBL. Information was passed between the two regions of LBL in two particular ways. The expected value of the attentional nodes $E(y|x)$ in the first region given the stimulus cues was passed

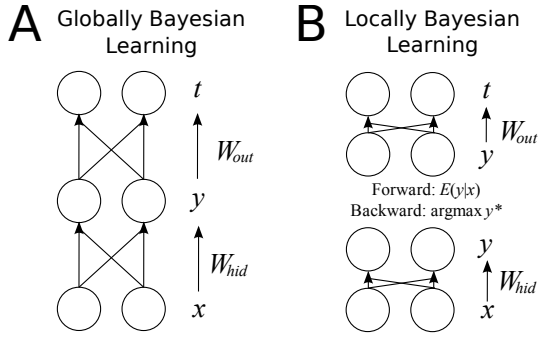

Figure 1: Network diagrams of Locally and Globally Bayesian Learning. The underlying models are feedforward neural networks with input vector x, hidden weight matrix W_h, hidden node vector y, output weight matrix W_o, and output vector t. In GBL, all of the hidden parameters are inferred together. In LBL, the network graph has been sectioned into regions with copies of the hidden nodes y in each region. Messages are passed back and forth between the copies of the hidden nodes y, the expected value $E(y|x)$ is passed upward and the single $y*$ that best maximizes the output is passed backwards.

to the second region, and computations depending on y in the second region were calculated based on $E(y|x)$. Once the feedback response t was received by the system, the output weights W_{out} were updated to be $p(W_{out}|E(y|x),t)$, instead of the $p(W_{out}|x,t)$ as they would be under GBL.

The second approximation was in the information passed back from the second region to the first region. In this approximation the value of y that maximizes the probability of the feedback response t was computed,

$$\hat{y} = \arg\max_{y^*} \sum_{W_{out}} p(t|W_{out},y^*)p(W_{out}|E(y|x),t) \qquad (2)$$

and finally the hidden weight prior $p(W_{hid})$ was updated with $p(W_{hid}|x,\hat{y})$.

The local computation and message passing were motivated from several perspectives, including the desire to simplify calculations (Kruschke, 2006b). The use of local computation and message passing to simplify computation has been echoed in computer science and statistics, and next we describe a version of this approach and connect it to LBL.

Belief Propagation

Dividing a model into components that perform local updating, based only on the information available from contiguous components, is an idea explored in machine learning and statistics under the name Belief Propagation (BP; Minka, 2001; Yedidia, Freeman, & Weiss, 2005). BP is a message passing scheme for inference. The joint distribution is divided into regions, messages are passed between regions, and updating is performed based on these messages. Reducing

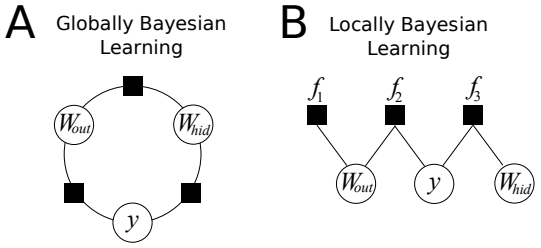

Figure 2: Factor graphs of the variables in the Globally and Locally Bayesian Learning. In A, the factor graph of GBL is a loop, while in B, the factor graph of LBL is a chain.

computation for Bayesian models is one of its main applications.

Typically, inference by message passing assumes that the distribution of interest is represented as a product of factor functions $f_a(\cdot)$

$$p(\mathbf{s}) = \frac{1}{Z}\prod_a f_a(\mathbf{s}_a) \qquad (3)$$

where $\mathbf{s}$ denotes a set of random variables and $\mathbf{s}_a$ the subset of variables that are arguments to the function $f_a(\cdot)$. Constant Z is the normalizing constant of the distribution.

BP schemes are most commonly used as approximations to marginals of distributions. In these schemes, one defines a set of *belief functions* $b_r(\mathbf{s}_r)$ that will encode marginals of a subset of variables $\mathbf{s}_r \subseteq \mathbf{s}$: that is, belief functions are probability functions, being non-negative and integrating to one. The goal is to project the marginals of the (exact) distribution $p(hidden \mid observed)$ into the corresponding belief functions in a iterative and computationally cheap way – a brute-force procedure that computes these marginals directly is in general not feasible.

In most cases, the belief functions will only be approximations to the true marginals, and the set of belief functions do not need to be globally coherent (in the sense there exists a single joint distribution with marginals given by the belief functions). Message passing can be applied to many types of probabilistic models, but we focus on the special case that is relevant to the interpretation and generalization of LBL.

Message Passing for a Factored Prior

In the probabilistic model, the complete set of random variables used when observing the cue-target pair $\{x,t\}$ is given by $\{x,y,t,W_{hid},W_{out}\}$. When a cue-target pair $\{x,t\}$ is revealed, Bayesian updating will consist of computing $p(W_{hid},W_{out},y \mid x,t)$. Within LBL and GBL, the posterior can be factored similarly to Equation 1. Using the framework of Equation 3, the posterior is the result of multiplying (and renormalizing) these factors,[1]

[1]There are multiple ways to construct the factor graph for these variables. We could have instead treated the product $f_1(W_{out})f_2(W_{out},y)$ as a single factor, but this version better matches LBL.

$$\begin{aligned} f_1(W_{out}) &\equiv p(W_{out}) \\ f_2(W_{out}, y) &\equiv p(t \mid W_{out}, y) \\ f_3(W_{hid}, y) &\equiv p(y \mid W_{hid}, x) p(W_{hid}) \end{aligned} \tag{4}$$

Notice that we are considering stimulus $\{x,t\}$ as fixed, as such, they are not arguments to the factor functions. We also assume that the prior for the parameters factorizes as $p(W_{out})p(W_{hid})$ at any stage of our experiment, an assumption we will justify in the next section.

Consider a *chained* propagation scheme for defining beliefs. The LBL graph of Figure 2b depicts our target distribution using blocks to denote factors and circles to denote variables, a representation also known as a *factor graph* (Yedidia et al., 2005). In constrast, GBL might also have as a prior a function $p(W_{out}, W_{hid})$ for all data points. This implies that a factor between W_{out} and W_{hid} is in general also needed, which results in the loop in Figure 2a.

In LBL, we define messages that will pass from variable nodes (the bottom, circled ones, in Figure 2b) to factor nodes (top, squared nodes), and vice-versa. As we will see, such messages will stand for different partial summations over the distribution function of interest, allowing us to compute marginals in a more efficient way than by brute-force summation. Consider the expression for the marginal of W_{out} given the evidence after we sum out the other hidden variables,

$$\begin{aligned} p(W_{out} \mid x,t) &\propto \sum_y \sum_{W_{hid}} f_1(W_{out}) f_2(W_{out}, y) f_3(W_{hid}, y) \\ &= f_1(W_{out}) \sum_y f_2(W_{out}, y) \sum_{W_{hid}} f_3(W_{hid}, y) \end{aligned} \tag{5}$$

The inner summation over W_{hid} can be cached as a function of the corresponding values of y. Expressing that as a message passing algorithm, we denote the value of this inner summation as a message. Since the message goes from factor f_3 to variable y and depends on the particular value of y, we denote it as $m_{f_3 \to y}(y)$ [2].

Node y repeats the message from f_3 to f_2, since it has no other neighbors but these two. We denote this copy as $m_{y \to f_2}(y)$. Given this message, a message from f_2 to W_{out} can be defined in terms of the summation over y using the cached message,

$$m_{f_2 \to W_{out}}(W_{out}) \propto \sum_y f_2(W_{out}, y) m_{y \to f_2}(y) \tag{6}$$

For simplicity of interpretation, we assume we normalize all messages – in general, this is not necessary. Notice that because we cached the previous summation, we are summing over a reduced space now. Message passing can be seen as a dynamic programming approach to probabilistic inference.

Finally, since factor node f_1 has no neighbors but W_{out}, it passes its factor function to W_{out} with no summation required: $m_{f_1 \to W_{out}} = f_1(W_{out})$. The marginal for W_{out} given $\{x,t\}$ can be rewritten in message passing notation as the belief function $b_{W_{out}}(W_{out})$,

$$b_{W_{out}}(W_{out}) \propto m_{f_1 \to W_{out}}(W_{out}) m_{f_2 \to W_{out}}(W_{out}) \tag{7}$$

In general, the belief function resulting from a message passing scheme will produce only an approximation to the desired marginal. The belief function would be approximate for GBL, because the GBL factor graph in Figure 2a contains a loop. This is not the case for the simple chain model of LBL in Figure 2b, which is guaranteed to calculate the correct marginal. As we just verified, $b_{W_{out}}(W_{out}) = p(W_{out} \mid x,t)$.

Calculating the marginal belief $b_{W_{hid}}(W_{hid})$ can be done by an analogous process starting from the other end of the chain (i.e., starting from message $m_{f_1 \to W_{out}}$). Computing the messages toward W_{hid}, our first summation comes as we pass information to node y,

$$m_{f_2 \to y}(y) \propto \sum_{W_{out}} f_1(W_{out}) f_2(W_{out}, y) \tag{8}$$

As before, the message from y to f_3 will again be just a copy of the incoming message from f_2. The message passing scheme will be finalized by the message from f_3 to W_{hid}, $m_{f_3 \to W_{hid}}$. This message will be the factor function f_3 weighted by the incoming message from y, marginalizing over y so that we obtain the marginal information for W_{hid},

$$m_{f_3 \to W_{hid}}(W_{hid}) \propto \sum_y m_{f_2 \to y}(y) f_3(W_{hid}, y) \tag{9}$$

Because there are no other messages into W_{hid}, its belief function $b_{W_{hid}}(W_{hid})$ and the posterior $p(W_{hid}|x,t)$ will be identical to the above message.

Approximations to Belief Propagation in Locally Bayesian Learning

We can now understand LBL algorithm as a belief propagation algorithm with three approximations. Perform the first stage as we described, passing messages from W_{hid} towards W_{out}. The first difference is that LBL replaces the message $m_{f_3 \to y}$ with the "collapsed" message,

$$m'_{f_3 \to y}(y) = \delta(y = E_{y \sim m_{f_3 \to y}}(y)) \tag{10}$$

where $E_{y \sim m_{f_3 \to y}}(y)$ is the expectation of y with respect to $m_{f_3 \to y}(y)$. Function $\delta(\cdot)$ is one if its argument is true, zero otherwise. The rest of the message passing scheme towards W_{out} stays the same, with the modified $b'_{W_{out}}(W_{out})$ being an approximation to $p(W_{out} \mid x,t)$.

The second and third approximations change the information used to update W_{hid}. Instead of using the prior distribution of W_{out} in the update of W_{hid}, LBL uses the posterior of W_{out} in the update. This approximation corresponds to an overcounting of the evidence in the updating of W_{hid}. The overcounting is reflected in the new message,

[2] In general message passing situations, we expect f_3 to first get a message from W_{hid}, but since node W_{hid} in Figure 2b has no other neighbors, this message is empty.

$$m'_{f_2\to y}(y) \propto \sum_{W_{out}} p(W_{out} \mid E(y|x),t)p(t \mid W_{out},y) \quad (11)$$

$$= \sum_{W_{out}} b'_{w_{out}}(W_{out})f_2(W_{out},y) \quad (12)$$

Instead of using the full message, LBL further approximates this distribution by collapsing it to its maximum value:

$$m''_{f_2\to y} = \delta(y = \hat{y}) \quad (13)$$

where $\hat{y} = \text{argmax}_{y^*} \sum_{W_{out}} b'_{w_{out}}(W_{out})f_2(W_{out},y^*)$. The message passing is finalized by (9) using $m''_{f_2\to y}$ instead of $m_{f_2\to y}$. LBL approximation algorithm is recovered, since the new message from f_3 to W_{hid} is now

$$m'_{f_3\to W_{hid}}(W_{hid}) \propto p(\hat{y} \mid W_{hid},x)p(W_{hid}) \quad (14)$$

which given the normalization will be exactly $p(W_{hid} \mid \hat{y},x)$.

In contrast, even if the prior in GBL is factored, its factor graph changes after processing the first data point. The dependency between W_{out} and W_{hid} will transform the graph from Figure 2b into the graph in Figure 2a, starting from the second data point. The side-effect of the collapsing done in Equation 13 is that W_{out} and W_{hid} will be marginally independent. This approximate posterior distribution is then used as the new prior for the next data point and the process repeated.

An Alternative Approximation

Once we understand LBL as a special case of a message passing algorithm in a chain-ordered factor graph, we can experiment with other approximations to the full Bayesian model. The consequence of the LBL approximations that we focus on is the independence that results from the collapsed messages. This independence can also be produced using the full messages, which is a type of Assumed Density Filtering approximation (ADF; Boyen & Koller, 1998). In ADF, message passing is used to approximate a posterior over parameters for a single data point, and this approximate posterior is used as the prior when processing the next point.

Instead of using the expected value $E(y|x)$ or the maximum y^* to remove dependencies between W_{out} and W_{hid}, we will adopt the standard procedure of ADF: considering all factorized distributions $q(W_{out},W_{hid}) \equiv q_o(W_{out})q_h(W_{hid})$, find the one that is closest to the true posterior $p(W_{out},W_{hid} \mid x,t) \equiv p_{x,t}(W_{out},W_{hid})$ according to the KL-divergence criterion,

$$KL(p \,||\, q) = \int p_{x,t}(W_{out},W_{hid}) \ln \frac{q_o(W_{out})q_h(W_{hid})}{p_{x,t}(W_{out},W_{hid})} dW_{out}W_{hid} \quad (15)$$

Minimizing Equation 15 with respect to $q_o(\cdot)$ and $q_h(\cdot)$ results in $q_o(W_{out}) = p(W_{out} \mid x,t)$ and $q_h(W_{hid}) = p(W_{hid} \mid x,t)$. In this way we obtain the exact marginals under independence, and use this approximation to the posterior as the prior when processing the next data point. The approximation used in LBL produces a non-standard projection of the true posterior into the space of factorized distributions, so the ADF using the exact marginals will always be as good or better than LBL in the KL divergence sense.

Updating the parameters using ADF is accomplished by using the complete messages from Equations 4-9 in place of the approximated versions used in LBL. As in the LBL, learning is the result of local computations and message passing between regions.

Highlighting

We have connected LBL to ADF and BP and have proposed an alternative model based on this formulation. Now we can explore how well these models predict human behavior in a conditioning experiment. Space constraints prevent us from exploring the range of conditioning effects LBL has been applied to, so we focus instead on the most troublesome effect for Bayesian models, highlighting.

Highlighting (Kruschke, 1996, 2006b; Medin & Edelson, 1988) is a conditioning effect that results from exposing subjects to two types of trials. The first, which is notated at $I.PE \to E$, presents the subject with two cues: cue I and cue PE. Following the cues is the outcome E. The second type of trial is $I.PL \to L$, in which subjects are trained on cues I and PL and which leads to outcome L. In this task, cue I is an imperfect predictor of the outcomes, while cues PE and cue PL are perfect predictors of E and L respectively.

The highlighting effect occurs in designs in which subjects initially learn $I.PE \to E$ trials, and later learn $I.PL \to L$ trials. When presented at test with cue I, subjects tend to choose outcome E. But when presented at test with cues PE and PL, subjects tend to choose outcome L. This effect is usually explained in terms of attention: subjects first learn to associate outcome E equally with cues I and cues PE, because they are both equally predictive of this outcome in the $I.PE \to E$ trials. However, when later learning the $I.PL \to L$ trials trials, subjects realize the cue I is not informative, so ignore it and heavily weight the association between cue PL and outcome L. During test, I has a stronger association with outcome E and when cues PE and PL compete against each other, outcome L is chosen because PL has a stronger association to L than PE has to E.

Kruschke (2006a, 2006b) demonstrated that highlighting was an extremely challenging effect for Bayesian models of conditioning. The key challenge to these models is the *canonical design*: an equal number of $I.PE \to E$ and $I.PL \to L$ trials over the entire experiment. The design that accomplishes the balance between global trial equality and local trial imbalance is shown in Table 1. Using this design, the prediction from a stationary Bayesian model is indifference to the outcomes for both cue I and cues $PE.PL$. However both human data (Kruschke, in press) and LBL (Kruschke, 2006b) predict a robust highlighting effect.

We show the predictions of GBL and LBL for highlighting in Figure 3 as well as the results for human subjects estimated

Table 1: Kruschke's Canonical Highlighting Design

Phase	Number of Trials	Items
First	$2*N_1$	$I.PE \rightarrow E$
Second	$3*N_2$	$I.PE \rightarrow E$
	$1*N_2$	$I.PL \rightarrow L$
Third	$1*(N_2+N_1)$	$I.PE \rightarrow E$
	$3*(N_2+N_1)$	$I.PL \rightarrow L$

in Kruschke (in press). In the first three panels, the model was trained on the same stimuli used in Kruschke (2006b): 7 trials of $I.PE \rightarrow E$ followed by 7 trials of $I.PL \rightarrow L$. GBL shows no highlighting effect because of its insensitivity to order. LBL shows a highlighting effect that is almost exactly the size of human effect. ADF shows a highlighting effect that is between GBL and LBL, but is not as large as in the human data.

However, the human data were collected using many more trials, $N_1 = 10$ and $N_2 = 5$ in the canonical design resulting in 50 trials of each type[3] . When we train LBL and ADF on the number of trials used in Kruschke (in press), then both models produce highlighting effects as large or larger than human highlighting effects (Figure 3).

Predictions for Alternating Trials

The highlighting effect is assumed to be caused by early blocks with a high proportion of $I.PE \rightarrow E$ trials and late blocks with a high proportion of $I.PL \rightarrow L$ trials. An extreme example of the highlighting design is alternating trials, in which single $I.PL \rightarrow L$ trial is followed by a $I.PE \rightarrow E$ trial. The early block and late block are both one trial long and there can be many repetitions of the blocks. As far as we are aware, the highlighting effect has not been tested for in this design, and it would be surprising if it were found. A highlighting effect for alternating stimuli would mean that subjects are extremely sensitive to small changes in order, in fact, sensitive to the position of a single trial in the entire run of the experiment. Shifting the first trial to the end of the experiment in this design produces the opposite alternating order.

Figure 4 shows the predictions for the models on the alternating trials. As in the highlighting design, GBL predicts equal preference for the two alternatives when test with cue I or with cues $PE.PL$. For both 14 and 100 trials, LBL predicts a highlighting effect that is as strong as the effect predicted for the highlighting stimuli. ADF closely mimics GBL and predicts nearly indifferent performance for both 14 and 100 alternating trials.

Discussion

Our paper provides a connection between LBL and approximations in computer science and statistics and uses this connection to propose an alternative to LBL. Casting LBL in the framework of BP identified several deviations from this message passing algorithm. The deviations cause the posterior distribution for the weights to be factorized. We use ADF to retain this factorization of the posterior weights, while producing the best possible approximation (as measured by KL divergence) to the full posterior distribution. Like LBL, ADF uses local message passing and it provides an interpolation between the more computationally complex full Bayesian model and the more approximate LBL.

Highlighting has proven to be a difficult experimental effect to predict using Bayesian models (Daw et al., 2008; Kruschke, 2006a, 2006b) and LBL predicts these data, as does our alternative, ADF. The differences in the approximation are revealed, however, when they are trained on alternating highlighting trials. ADF predictions are very close to the full Bayesian model, while the stronger approximations in LBL continue to predict a strong highlighting effect. This result may only hold if learning is applied to the parameter spaces used in (Kruschke, 2006b), as a preliminary investigation using larger hypothesis spaces shows that LBL does not produce highlighting for alternating trials (Kruschke & Denton, 2009). Further investigations are needed to test the predictions of these approximations against human highlighting effects.

In addition to highlighting, LBL produces human-like behavior in a variety of conditioning paradigms. We have focused on the highlighting effect in this paper due to space constraints. Our initial simulations do show that ADF is able to produce forward blocking, backward blocking, and a larger effect for forward blocking than backward blocking.

Conclusions

Kruschke (2006b) introduced the idea that conditioning effects, such as highlighting, could be produced by using local message passing to approximate full Bayesian models. Our work builds on this approach by connecting it to message passing algorithms used in computer science and statistics, and develops an alternative that can also predict highlighting behavior. Connections between existing models and machine learning algorithms give cognitive scientists access to a rich resource for developing alternative models that produce a range of behavior. Through matching these models to behavior it is hoped that the approximations used in the mind can be determined.

Acknowledgments

Adam Sanborn was supported by a Royal Society USA Research Fellowship and the Gatsby Charitable Foundation.

References

Anderson, J. R. (1991). The adaptive nature of human categorization. *Psychological Review*, *98*(3), 409–429.

Boyen, X., & Koller, D. (1998). Tractable inference for complex stochastic processes. In *Proceedings of the fourteenth*

[3] As in Kruschke (in press), a canonical highlighting design was used, but to simplify the results the simulations used only a single copy of the canonical design.

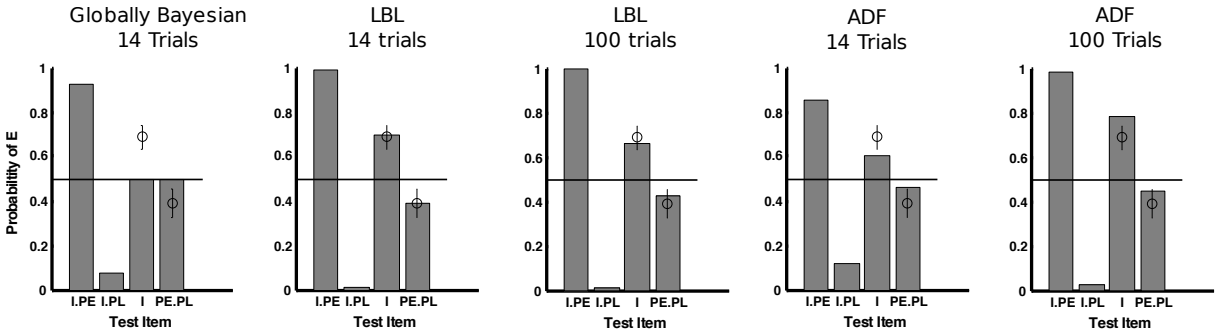

Figure 3: Highlighting results for a selection of models and experimental designs compared to human data. The mean human preference is plotted on each graph with a circle. Bars around the circle to show 95% confidence intervals on the human data. The bar plots show the model predictions of outcome E, where the line marks equal preference between outcomes E and L. A standard set of cues is tested in each model: the original training sets of cues $I.PE$ and $I.PL$, as well as the critical tests of cue I and cues $PE.PL$. The set of models and stimuli used to train the models are described in the text.

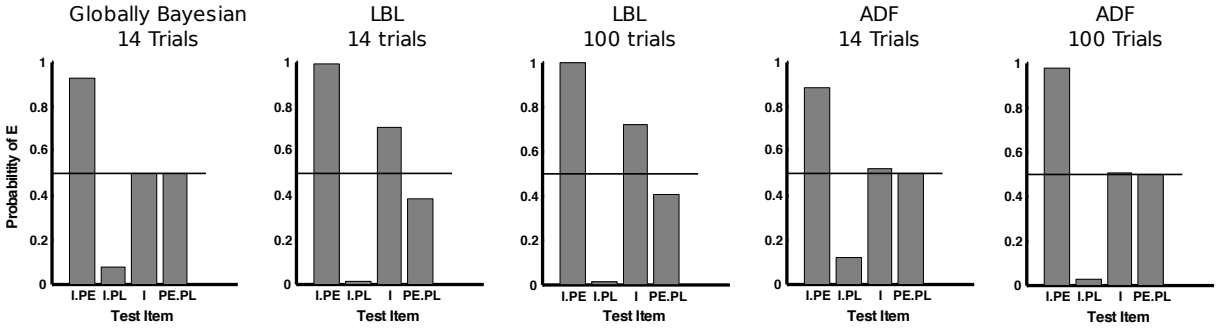

Figure 4: Alternating trial predictions for a selection of models and experiment lengths. The bar plots show the predictions of various models of outcome E, where the line marks equal preference between outcomes E and L. The original training cues $I.PE$ and $I.PL$, as well as the critical tests of cue I and cues $PE.PL$. The set of models and stimuli used to train the models are described in the text.

conference on uncertainty in artificial intelligence (p. 33-42).

Daw, N. D., Courville, A. C., & Dayan, P. (2008). Semi-rational models of conditioning: the case of trial order. In N. Chater & M. Oaksford (Eds.), *The probabilistic mind* (p. 431-452). Oxford, UK: Oxford University Press.

Gigerenzer, G., & Todd, P. M. (1999). *Simple heuristics that make us smart*. Oxford: Oxford University Press.

Kahneman, D., Slovic, P., & Tversky, A. (Eds.). (1982). *Judgment under uncertainty: Heuristics and biases*. Cambridge: Cambridge University Press.

Körding, K., & Wolpert, D. M. (2004). Bayesian integration in sensorimotor learning. *Nature*, *427*, 244-247.

Kruschke, J. K. (1996). Base rates in category learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *22*, 3-26.

Kruschke, J. K. (2006a). Locally Bayesian learning. In *Proceedings of the 28th annual meeting of the cognitive science society.* Hillsdale, NJ: Erlbaum.

Kruschke, J. K. (2006b). Locally Bayesian learning with applications to retrospective revaluation and highlighting. *Psychological Review*, *113*, 677-699.

Kruschke, J. K. (in press). Attentional highlighting in learning: a canonical experiment. In B. Ross (Ed.), *The psychology of learning and motivation.*

Kruschke, J. K., & Denton, S. (2009). personal communication.

Medin, D. L., & Edelson, S. M. (1988). Problem structure and the use of base-rate information from experience. *Journal of Experimental Psychology: General*, *117*, 68-85.

Minka, T. (2001). *A family of algorithms for approximate Bayesian inference*. Unpublished doctoral dissertation, MIT, Boston.

Sanborn, A. N., Griffiths, T. L., & Navarro, D. J. (2006). A more rational model of categorization. In *Proceedings of the 28th Annual Conference of the Cognitive Science Society.* Mahwah, NJ: Erlbaum.

Yedidia, J. S., Freeman, W. T., & Weiss, Y. (2005). Constructing free energy approximations and generalized belief propagation algorithms. *IEEE Transactions on Information Theory*, *51*, 2282–2312.